

# The Simulation-based High-efficiency Advanced Reactor Prototyping (SHARP) Framework

## Contact

- Tim Tautges, ANL

## Constituent Packages

- MOAB unstructured mesh
- Nek thermalhydraulics
- UNIC neutronics

## Bibliographic Reference

- A. Siegel, T. Tautges, A. Caceres, D. Kaushik, and P. Fischer, Software Design of SHARP, Joint International Topical Meeting on Mathematics & Computation and Supercomputing in Nuclear Applications (M&C + SNA 2007), Monterey, California, April 15-19, 2007.

## Links

- SHARP short course slides, [http://www.mcs.anl.gov/~tautges/downloads/sharp\\_moab\\_shortcourse.pdf](http://www.mcs.anl.gov/~tautges/downloads/sharp_moab_shortcourse.pdf)
- Software Design of SHARP, [http://www.mcs.anl.gov/~kaushik/Papers/siegel\\_mnc\\_2007.pdf](http://www.mcs.anl.gov/~kaushik/Papers/siegel_mnc_2007.pdf)
- MOAB wiki page, <http://trac.mcs.anl.gov/projects/ITAPS/wiki/MOAB>
- CGM (geometry) wiki page, <http://trac.mcs.anl.gov/projects/ITAPS/wiki/CGM>
- MOAB-SD: Integrated Structured and Unstructured Mesh Representation, <http://dx.doi.org/10.1007/s00366-004-0296-0>, or <http://www.springerlink.com/content/n0qdl0axhu38m50h/fulltext.pdf>
- ITAPS web site, <http://itaps-scidac.org>

## Putting things in Perspective

John H Magerlein wrote:

Tim, I'm afraid that I'm really confused about a number of things after reading the information you send yesterday and various links on the web. Below is a series of questions which I'd like to discuss with you. I'd really appreciate it if we can talk by phone on Thursday or you can reply by email. Thanks. \* I don't understand the relationship among ITAPS, CGM, MOAB, CUBIT, and SHARP. Can you explain further?

Sure.

ITAPS is an effort to define common interfaces to geometry, mesh, and a few other types of data, and to develop mesh-related tools based on those interfaces.

CGM and MOAB are my implementations of the iGeom and iMesh interfaces, respectively. There are a few other implementations of iMesh, but for the most part none of iGeom.

CUBIT is a mesh generation toolkit developed at Sandia National Labs. CUBIT uses CGM as its geometry capability (or, to be more precise, CGM was taken out of cubit then re-integrated in component form). CUBIT is not open source, but is available to those working under government contract (see [www.cubit.sandia.gov](http://www.cubit.sandia.gov) for more details). I am developing a set of smaller, more component-based meshing tools, to be used in conjunction with or separately from cubit. This effort is in the very early stages.

SHARP is a project at ANL to do coupled reactor simulation. The approach we're taking is to access geometry and mesh data through common interfaces, i.e. using the ITAPS interfaces. Infrastructure tools like mapping solution results between meshes are also being implemented based on these interfaces (though in a few cases we have to go all the way down to MOAB directly). The so-called "SHARP framework" includes CGM, MOAB, and these infrastructure tools. I say so-called because it's really more of a collection rather than a monolith.

\* If I understand what you said and what I read on the ITAPS web site, ITAPS is merely an interface specification. Am I correct that there is no code implementing ITAPS other than CGM and MOAB? However in your notes, you say things such as "Use ITAPS from Fortran & where memory isn't important, otherwise MOAB". I thus infer that there is some other implementation of ITAPS. Can you clarify?

First, there are other implementations of iMesh (the ITAPS mesh interface), but none of iGeom (specifically, there is one other one, but it's based entirely on a mesh representation, at least as released in the public domain).

When I talk about using ITAPS vs. MOAB, I'm referring to the interface functions.

\* You say that MOAB only interfaces with C++. Is it not correct that many of the codes of interest are in Fortran or C? If so, how will we be able to use those codes with MOAB?

Through iMesh, the ITAPS mesh interface. Most of the functionality available in MOAB can be accessed through iMesh, and for what can't be, we'll have to construct specific wrapper functions. In my experience with SHARP, much of the driver code is in C++, which then calls into C- or Fortran-based physics modules, so this usually isn't an issue.

\* I can find almost nothing about SHARP. Your tutorial "SHARP Framework Tutorial" seems to discuss ITAPS and MOAB at some length, but it says little about SHARP. Is there some documentation or description?

No documentation at the high level yet (we're working on the code first :) ). There are two physics codes, UNIC (neutronics) and Nek (TH), which are being developed as both standalone codes and modules which can be called from a coupled code. It's the coupled code which we refer to as sharp. This is in very early stages of development compared to Nek and UNIC. Nek is available as open source, UNIC is not (since it probably has export control restrictions).

\* If I want to download some code and try it out, how should I begin? Is there any type of user interface and example data which I could use to get some idea that it is working?

You could download CGM and MOAB and build each of them. They don't have UI's, since they're mostly libraries. There are test applications for both included with the distributions. You should probably start with MOAB, since CGM will require a 3rd party package (OpenCascade?) which isn't small. MOAB includes some test meshes which

you could use to test small programs you wrote. MOAB's mbconvert utility can be used to convert these meshes to the vtk format, which can be read by Visit, Paraview, and other viz tools.